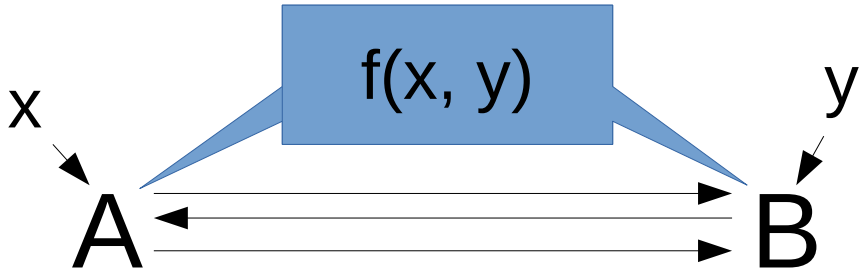
On the Structure of Unconditional UC Hybrid Protocols

Mike Rosulek (Oregon State University) and
Morgan Shirley (University of Toronto)

Problem Statement & Summary of Results

Our Parameters

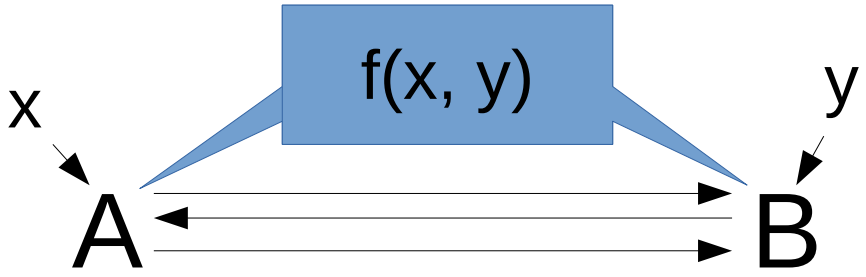
- 2-Party functions
- Finite and fixed truth tables
- Symmetric
- UC Security
- Security with abort
- Information theoretic



x \ y	0	1	2
0	0	1	2
1	1	2	0
2	2	0	1

Our Parameters

- 2-Party functions
- Finite and fixed truth tables
- Symmetric
- UC Security
- Security with abort
- Information theoretic




```
012
120
201
```

2-Party SFEs with Information-Theoretic UC Security

Either:

2-Party SFEs with Information-Theoretic UC Security

Either:

Trivial!

001
001

2-Party SFEs with Information-Theoretic UC Security

Either:

Trivial!

001
001

Impossible!

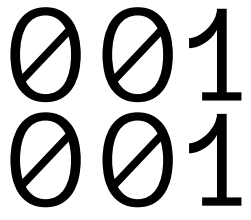
(literally everything interesting)

2-Party SFEs with Information-Theoretic UC Security

Either:

Trivial!

Impossible!

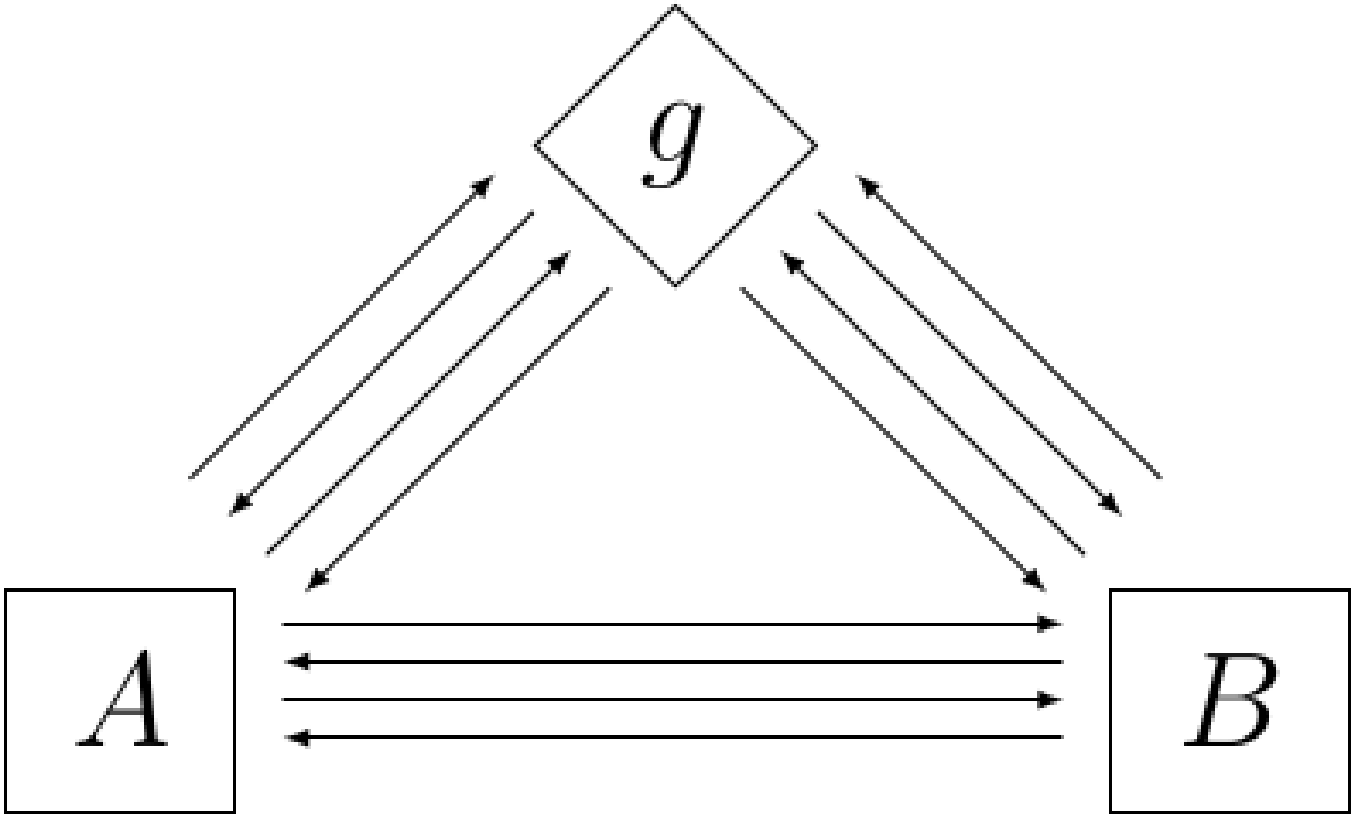


(literally everything interesting)

We'd like to differentiate functionalities on the right side

Canetti, Kushilevitz and Lindell *EUROCRYPT 2003*
Prabhakaran and Rosulek *CRYPTO 2008*

Hybrid World



Reductions

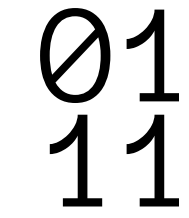
- A way to define complexity
- A function f **reduces to** a function g if there exists a ***g*-hybrid** protocol for f that has UC security

$$f \sqsubseteq g$$

Goal: completely classify when *f* and *g* reduce to each other

Completeness

- Complete *g*: every *f* reduces to *g*
- Kilian[1] shows a necessary and sufficient condition for completeness



1. In *23rd ACM STOC*, 1991

2-Party SFEs with Information-Theoretic UC Security

Trivial!

001
001

Reduces to everything

Complete

01
11

Everything reduces to

2-Party SFEs with Information-Theoretic UC Security

Trivial!

$$\begin{matrix} 001 \\ 001 \end{matrix}$$

Reduces to everything

Neither

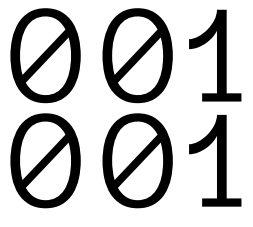
Complete

$$\begin{matrix} 01 \\ 11 \end{matrix}$$

Everything reduces to

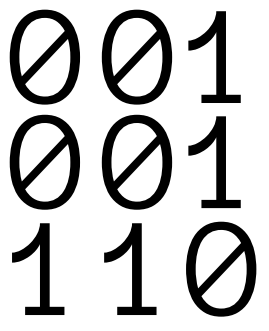
2-Party SFEs with Information-Theoretic UC Security

Trivial!



Reduces to everything

Neither



Complete

Everything reduces to

2-Party SFEs with Information-Theoretic UC Security

Trivial!

$$\begin{matrix} 0 & 0 \\ 1 & 1 \end{matrix}$$

Reduces to everything

Neither

$$\begin{matrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{matrix}$$

Some reductions studied between decomposable functions (e.g. Maji, Prabhakaran, Rosulek TCC 2009)

Complete

$$\begin{matrix} 0 \\ 1 \end{matrix}$$

Everything reduces to

2-Party SFEs with Information-Theoretic UC Security

Trivial!

$$\begin{matrix} 0 & 0 & 1 \\ 0 & 0 & 1 \end{matrix}$$

Reduces to everything

Neither

$$\begin{matrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{matrix} \qquad \begin{matrix} 1 & 1 & 2 \\ 4 & 5 & 2 \\ 4 & 3 & 3 \end{matrix}$$

Some reductions studied between decomposable functions (e.g. Maji, Prabhakaran, Rosulek TCC 2009)

Complete

$$\begin{matrix} 0 & 1 \\ 1 & 1 \end{matrix}$$

Everything reduces to

2-Party SFEs with Information-Theoretic UC Security

Trivial!

Neither

Complete

$$\begin{matrix} 0 & 0 & 1 \\ 0 & 0 & 1 \end{matrix}$$

Reduces to everything

$$\begin{matrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{matrix}$$

Some reductions studied between decomposable functions (e.g. Maji, Prabhakaran, Rosulek TCC 2009)

$$\begin{matrix} 1 & 1 & 2 \\ 4 & 5 & 2 \\ 4 & 3 & 3 \end{matrix}$$

?

$$\begin{matrix} 0 & 1 \\ 1 & 1 \end{matrix}$$

Everything reduces to

Main Theorem (almost)

When f and g are **incomplete**, if $f \sqsubseteq g$ then:

- $f \sqsubseteq g$ via a single-round deterministic protocol

Main Theorem (almost)

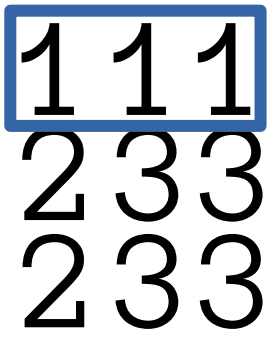
When f and g are **incomplete**, if $f \sqsubseteq g$ then:

- $f \sqsubseteq g$ via a single-round deterministic protocol

*With a few edge cases

Edge case: Unilateral functions

- At least one row (or column) constant!
- One party might know the output before the protocol begins



Main Theorem (almost)

When f and g are **incomplete**, if $f \sqsubseteq g$ then:

- $f \sqsubseteq g$ via a single-round deterministic protocol

Main Theorem (almost)

When f and g are **incomplete** and f is **non-unilateral**, if $f \sqsubseteq g$ then:

- $f \sqsubseteq g$ via a single-round deterministic protocol

Number of rounds required in a *g*-hybrid protocol for *f*

Number of protocol rounds necessary
(for incomplete and non-unilateral *f* and *g*)

Number of rounds required in a *g*-hybrid protocol for *f*

Number of protocol rounds necessary
(for incomplete and non-unilateral *f* and *g*)

Main Theorem (almost)

When f and g are **incomplete** and f is **non-unilateral**, if $f \sqsubseteq g$ then:

- $f \sqsubseteq g$ via a single-round deterministic protocol

Main Theorem

When f and g are **incomplete** and f is **non-unilateral**, if $f \sqsubseteq g$ via a (worst-case) log-round protocol:

- $f \sqsubseteq g$ via a single-round deterministic protocol

Main Theorem

When f and g are **incomplete** and f is **non-unilateral**, the following are equivalent:

- $f \sqsubseteq g$ via a (worst-case) log-round protocol
- $f \sqsubseteq g$ via a single-round deterministic protocol
- f **embeds** in g

Main Theorem

When f and g are **incomplete** and f is **non-unilateral**, the following are equivalent:

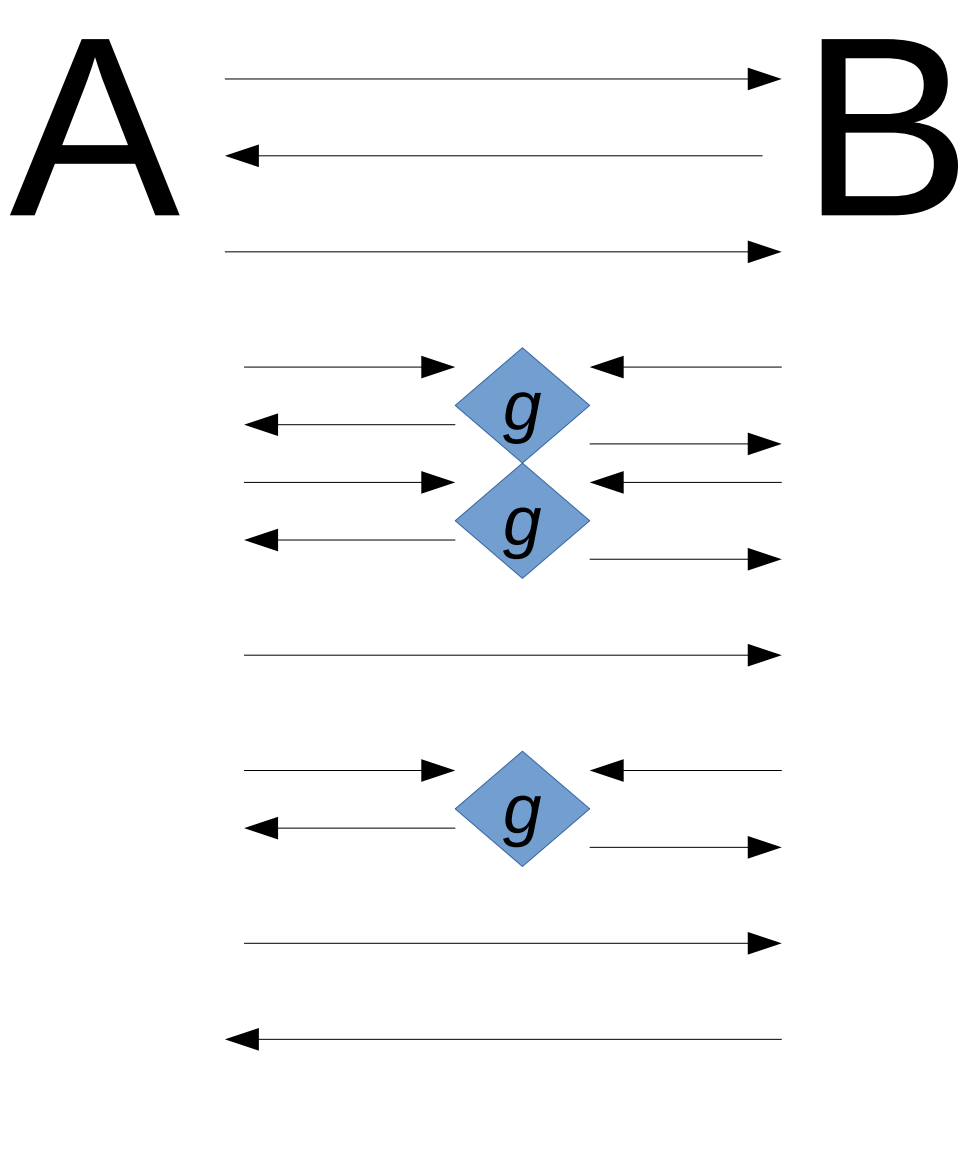
- $f \sqsubseteq g$ via a (worst-case) log-round protocol
- $f \sqsubseteq g$ via a single-round deterministic protocol
- f **embeds** in g

These edge cases are necessary

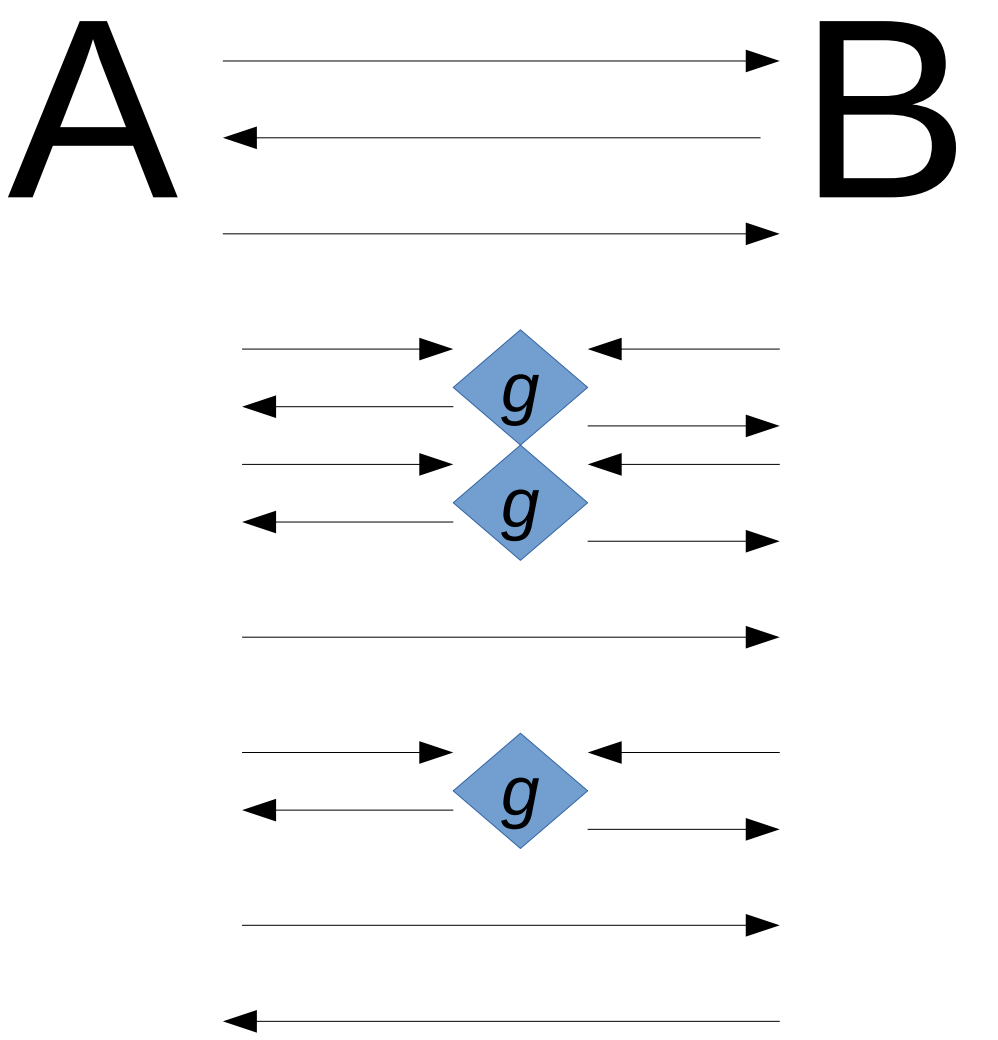
Embedding

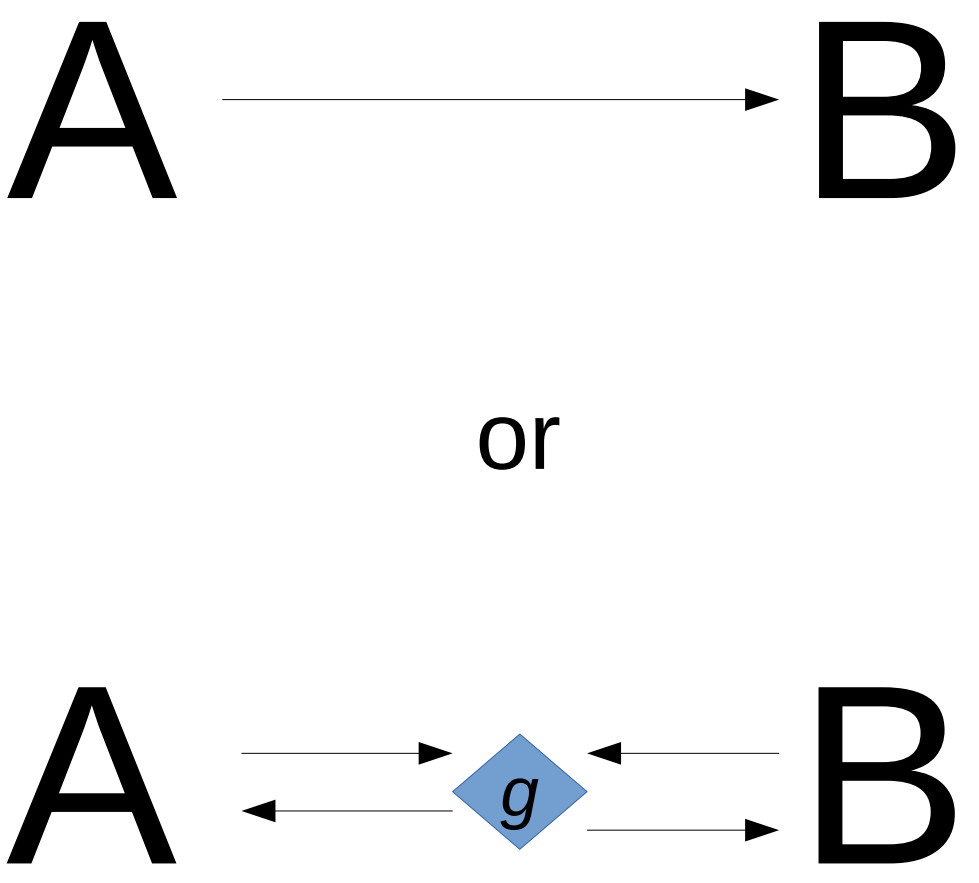
What would a single-round reduction look like?

What would a single-round reduction look like?

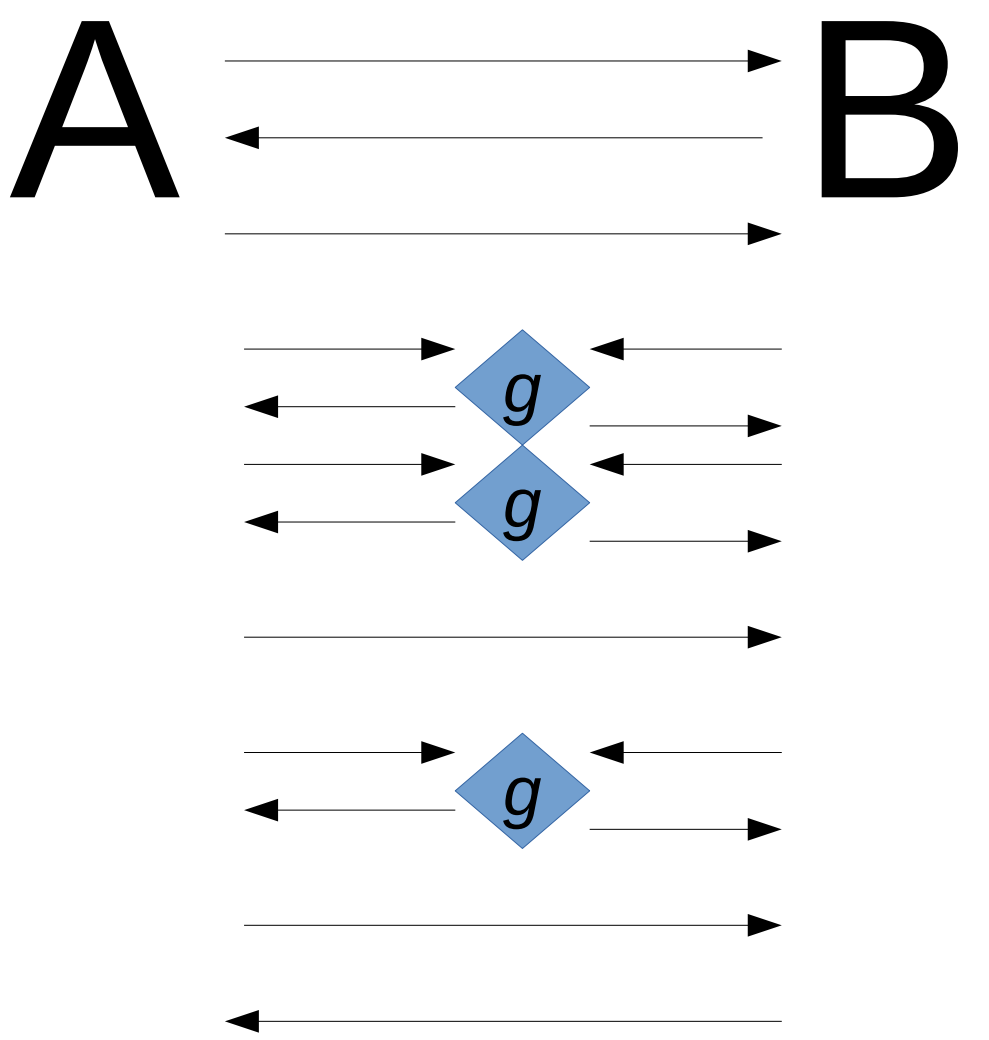


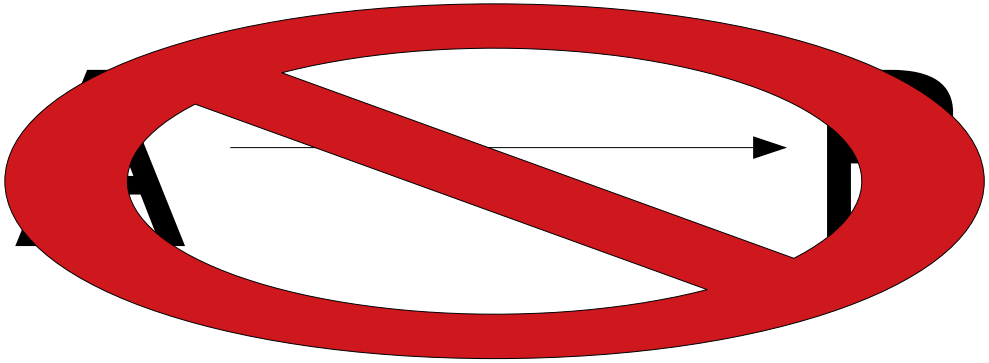
What would a single-round reduction look like?



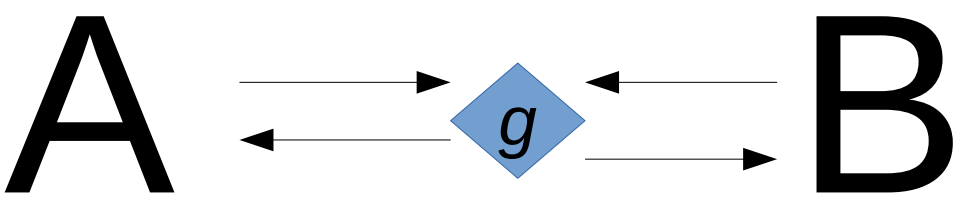


What would a single-round reduction look like?





or



Embedding: Correctness

- Each party sends a g-input based on their f-input
- The g-output maps back to an f-output
- Intuitively: f appears as sub-matrix* in g

f

1	1	2
4	5	2
4	3	3

g

1	1	2	7
4	5	2	7
4	3	3	7
8	8	8	9

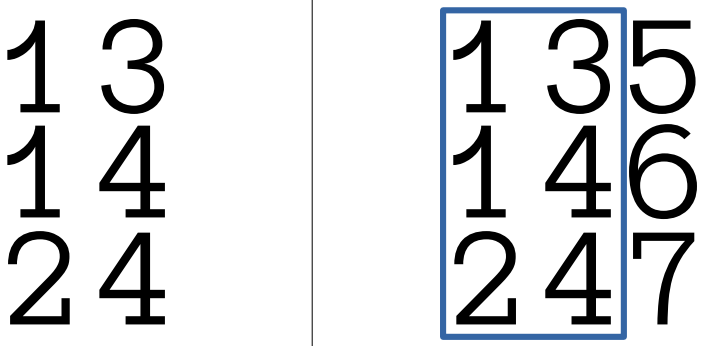
*Perhaps with some rearrangement and relabelling

Embedding: Security

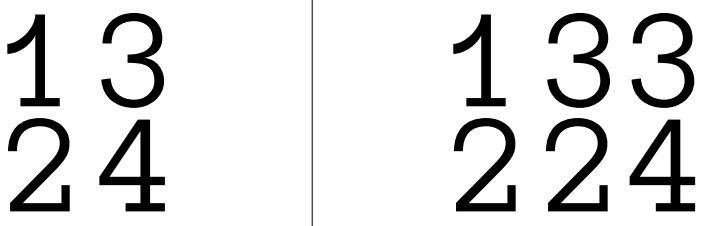
g can't reveal too much information	1 3 1 4 2 4	1 35 1 46 2 47
There are no ambiguous *g*-inputs	1 3 2 4	1 33 2 24

Embedding: Security

g can't reveal too much information



There are no ambiguous *g*-inputs



Embedding: Security

g can't reveal too much information

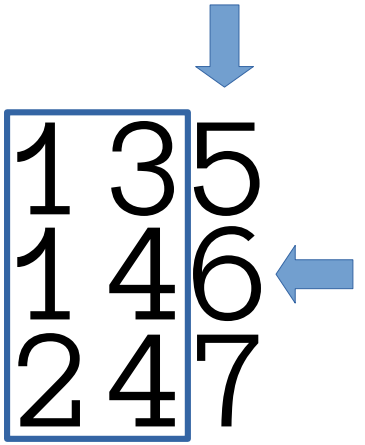
1 3 1 4 2 4	1 3 5 1 4 6 2 4 7

There are no ambiguous *g*-inputs

1 3 2 4	1 3 3 2 2 4

Embedding: Security

g can't reveal too much information

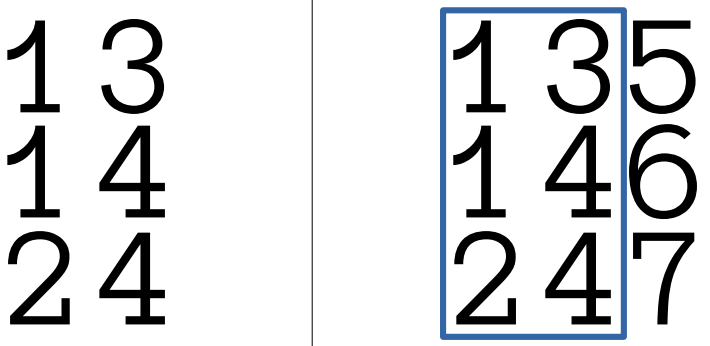
1 3 1 4 2 4	

There are no ambiguous *g*-inputs

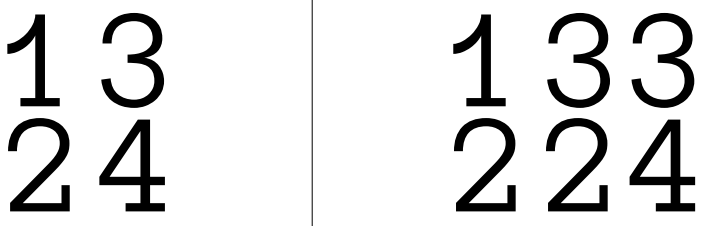
1 3 2 4	1 3 3 2 2 4

Embedding: Security

g can't reveal too much information

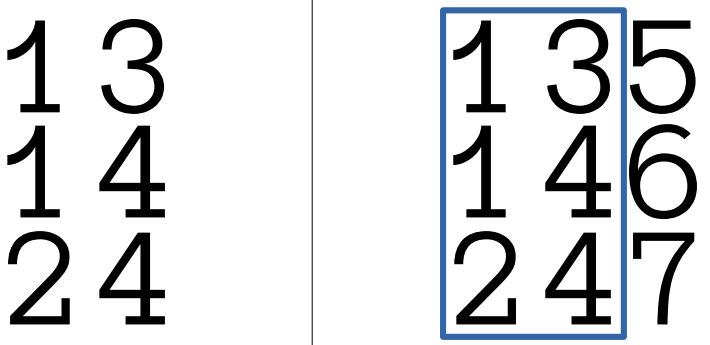


There are no ambiguous *g*-inputs

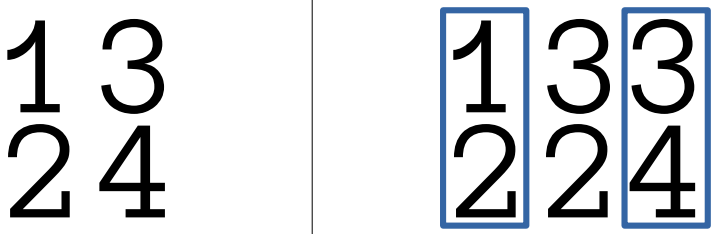


Embedding: Security

g can't reveal too much information



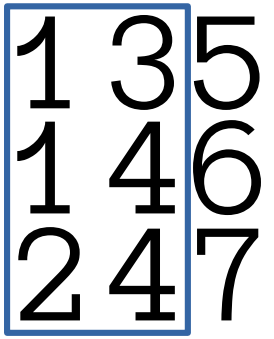
There are no ambiguous *g*-inputs



Embedding: Security

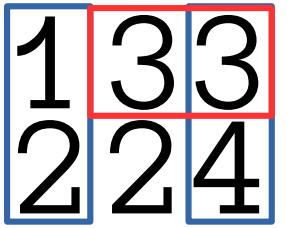
g can't reveal too much information

1 3
1 4
2 4



There are no ambiguous *g*-inputs

1 3
2 4



Embedding: Security

g can't reveal too much information

1	3
1	4
2	4

1	3	5
1	4	6
2	4	7

There are no ambiguous *g*-inputs

1	3
2	4

1	3	3
2	2	4

Embedding

- Definition basically follows this intuition
  - If there's an embedding, there's a single-round protocol
  - If there's a single-round protocol, there's an embedding

Main Theorem

When f and g are **incomplete** and f is **non-unilateral**, the following are equivalent:

- $f \sqsubseteq g$ via a (worst-case) log-round protocol
- $f \sqsubseteq g$ via a single-round deterministic protocol
- f embeds in g

Main Theorem

When f and g are **incomplete** and f is **non-unilateral**, the following are equivalent:

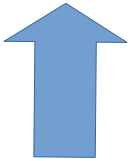
- $f \sqsubseteq g$ via a (worst-case) log-round protocol
- $f \sqsubseteq g$ via a single-round deterministic protocol

- f embeds in g

Main Theorem

When f and g are **incomplete** and f is **non-unilateral**, the following are equivalent:

- $f \sqsubseteq g$ via a (worst-case) log-round protocol

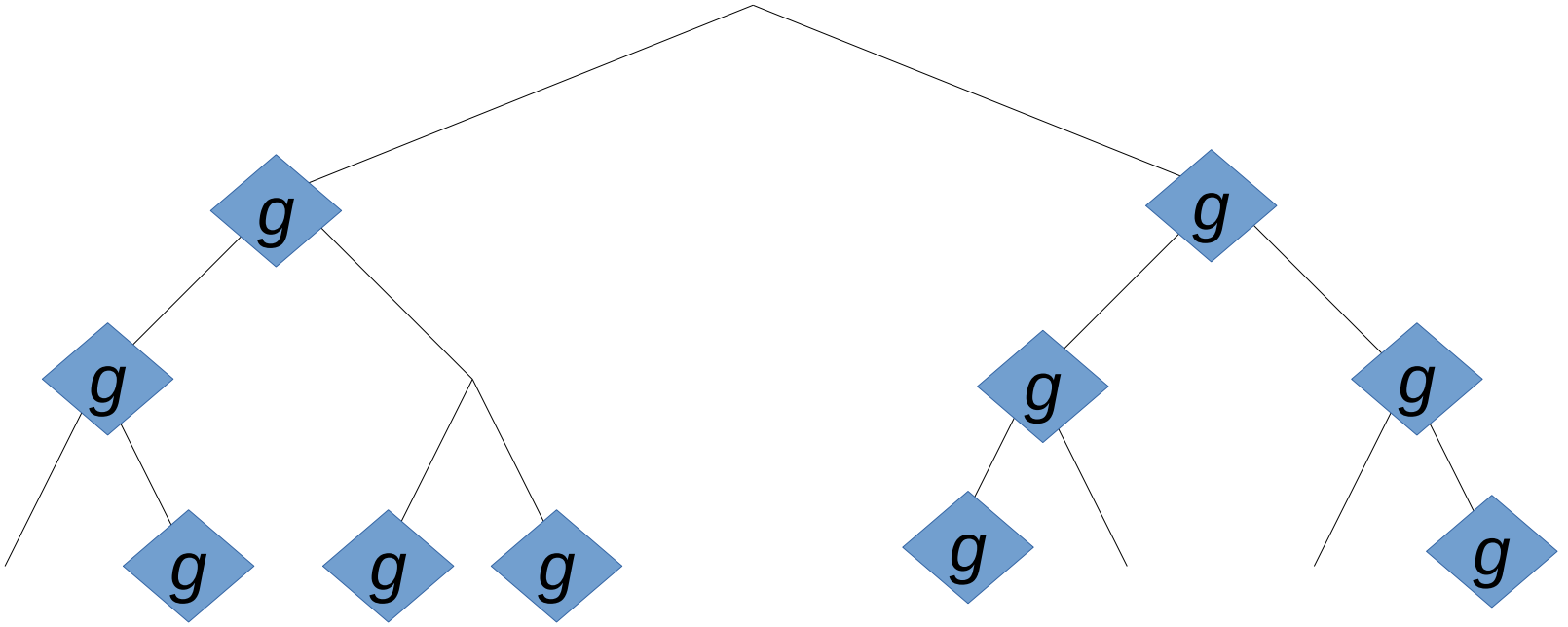


- $f \sqsubseteq g$ via a single-round deterministic protocol

- f embeds in g

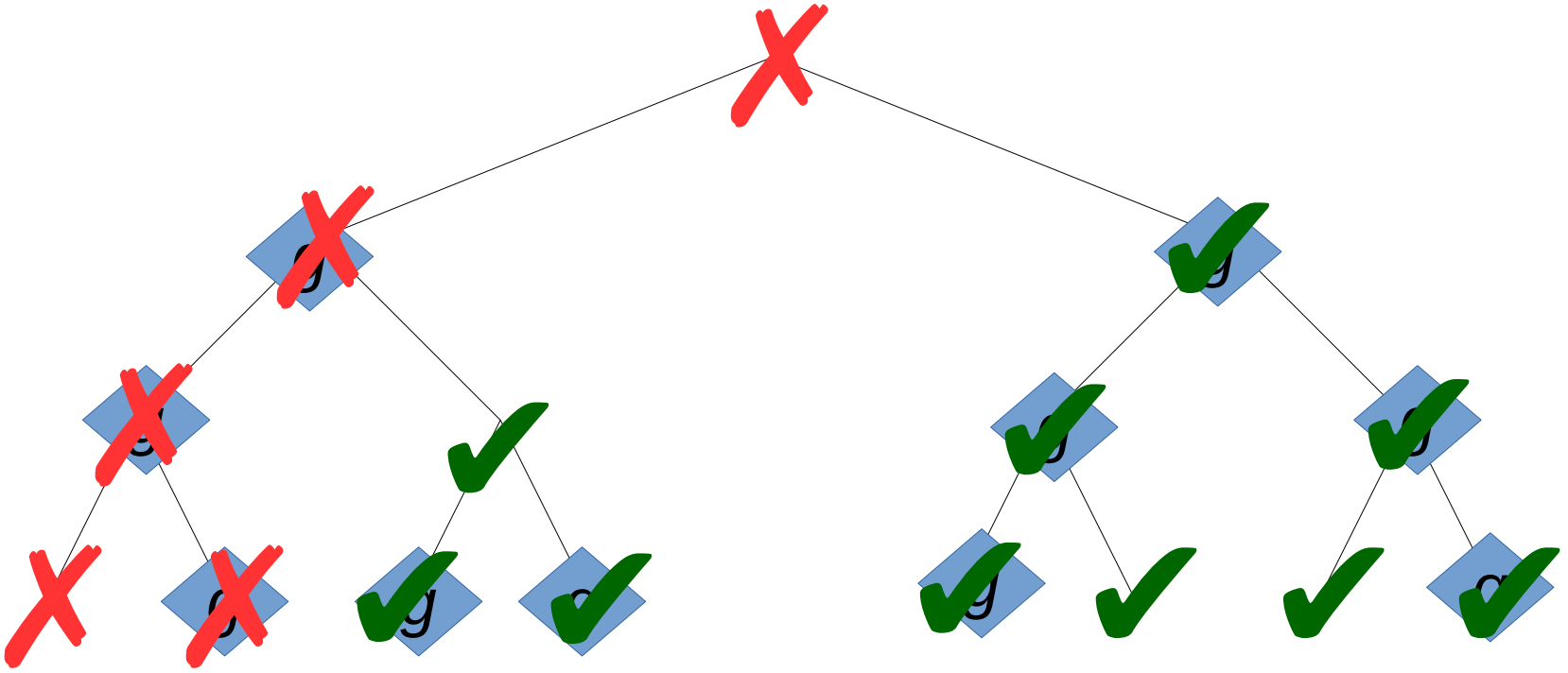
Collapse a protocol to a single round

Frontiers



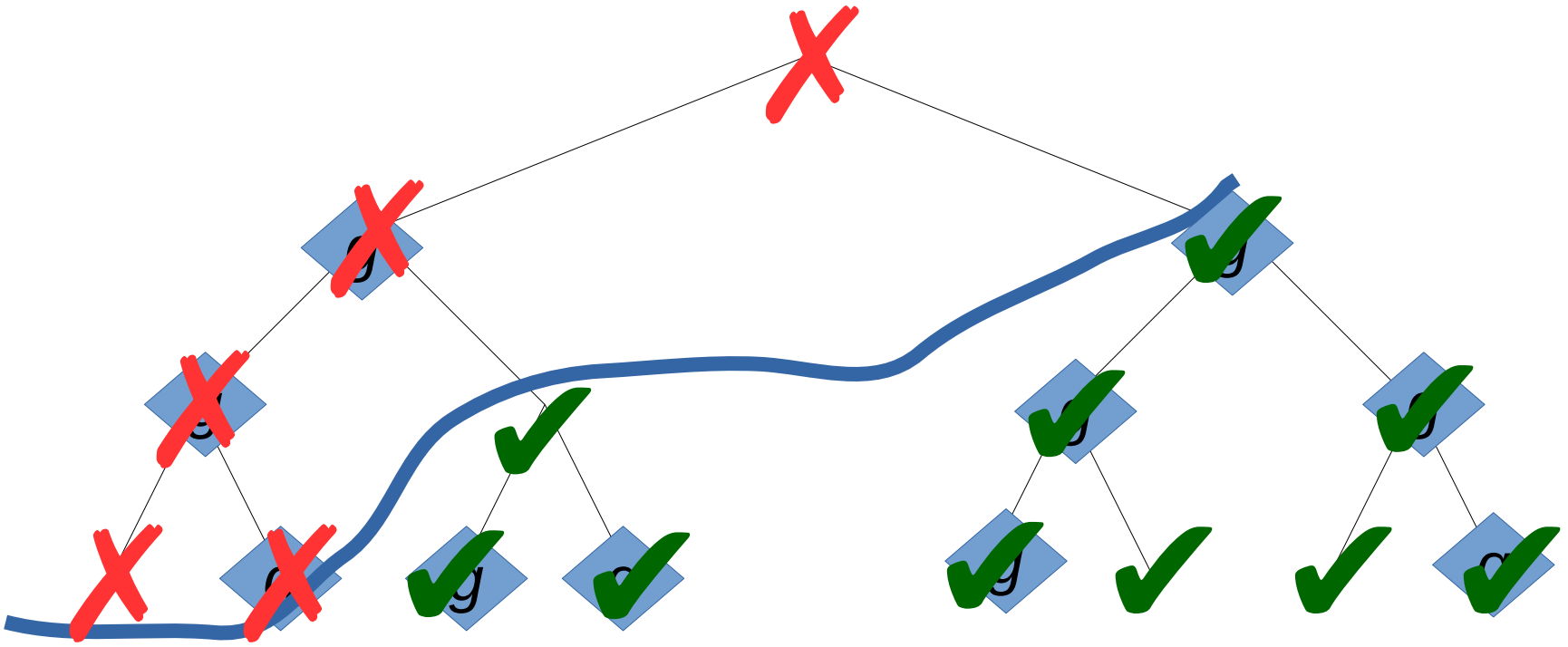
Frontiers

Property: Alice's simulator has extracted



Frontiers

Property: Alice's simulator has extracted



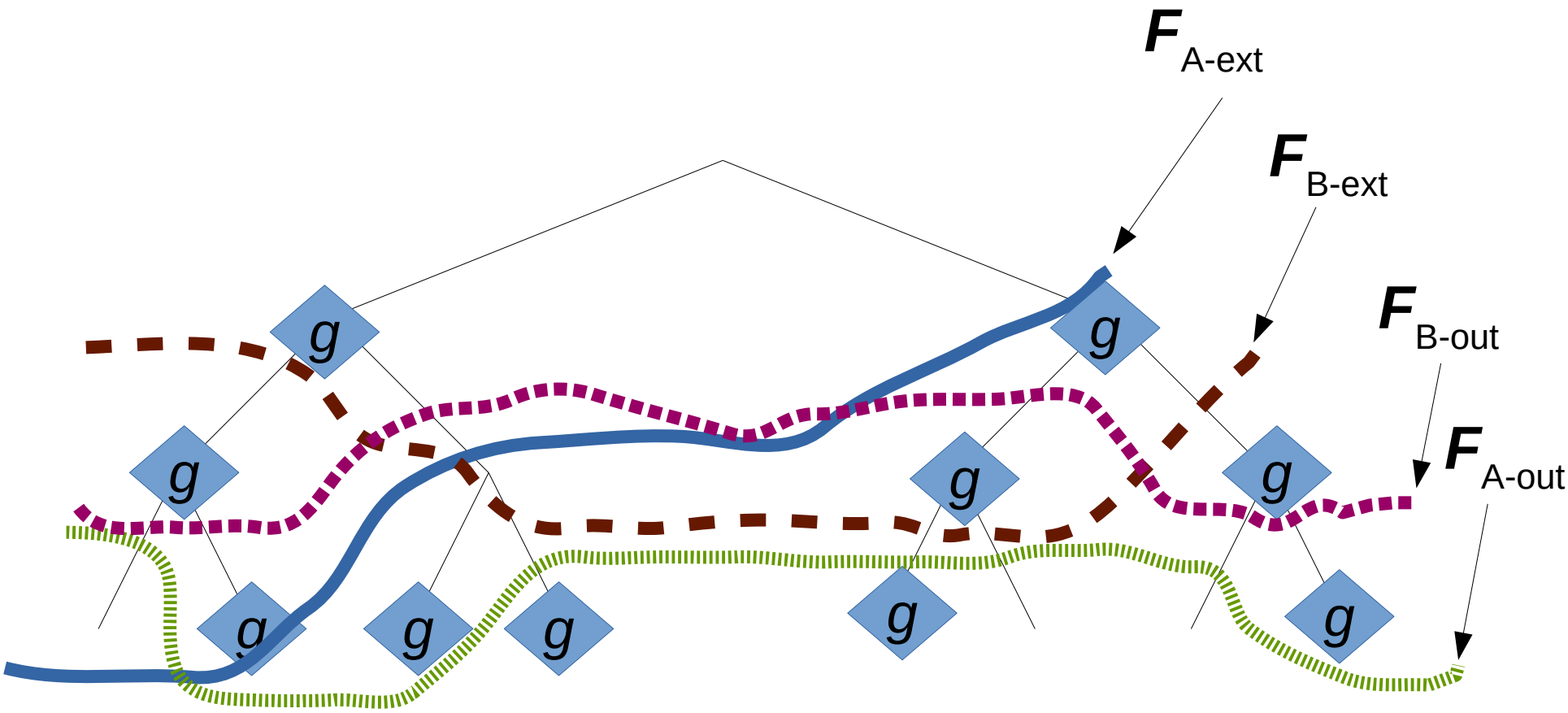
Our Frontiers

- $\boldsymbol{F}_{\text{A-ext}}$ – Alice's simulator has extracted
- $\boldsymbol{F}_{\text{A-out}}$ – Alice thinks the output is fixed (regardless of Bob's input)

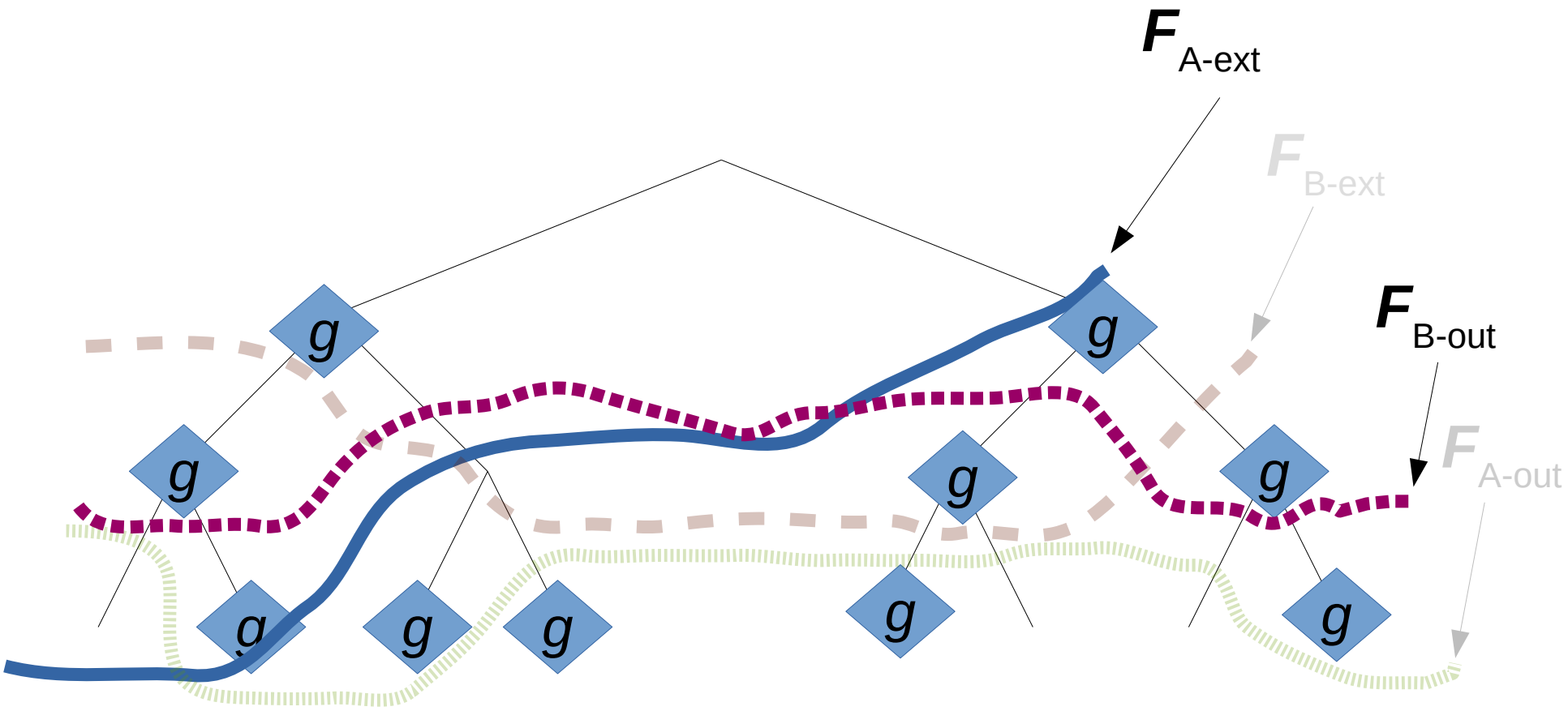
- Similar frontiers defined for Bob
  - $\boldsymbol{F}_{\text{B-ext}}$
  - $\boldsymbol{F}_{\text{B-out}}$

Idea: Give me any secure, correct protocol for $f \sqsubseteq g$. I can say something about the frontiers.

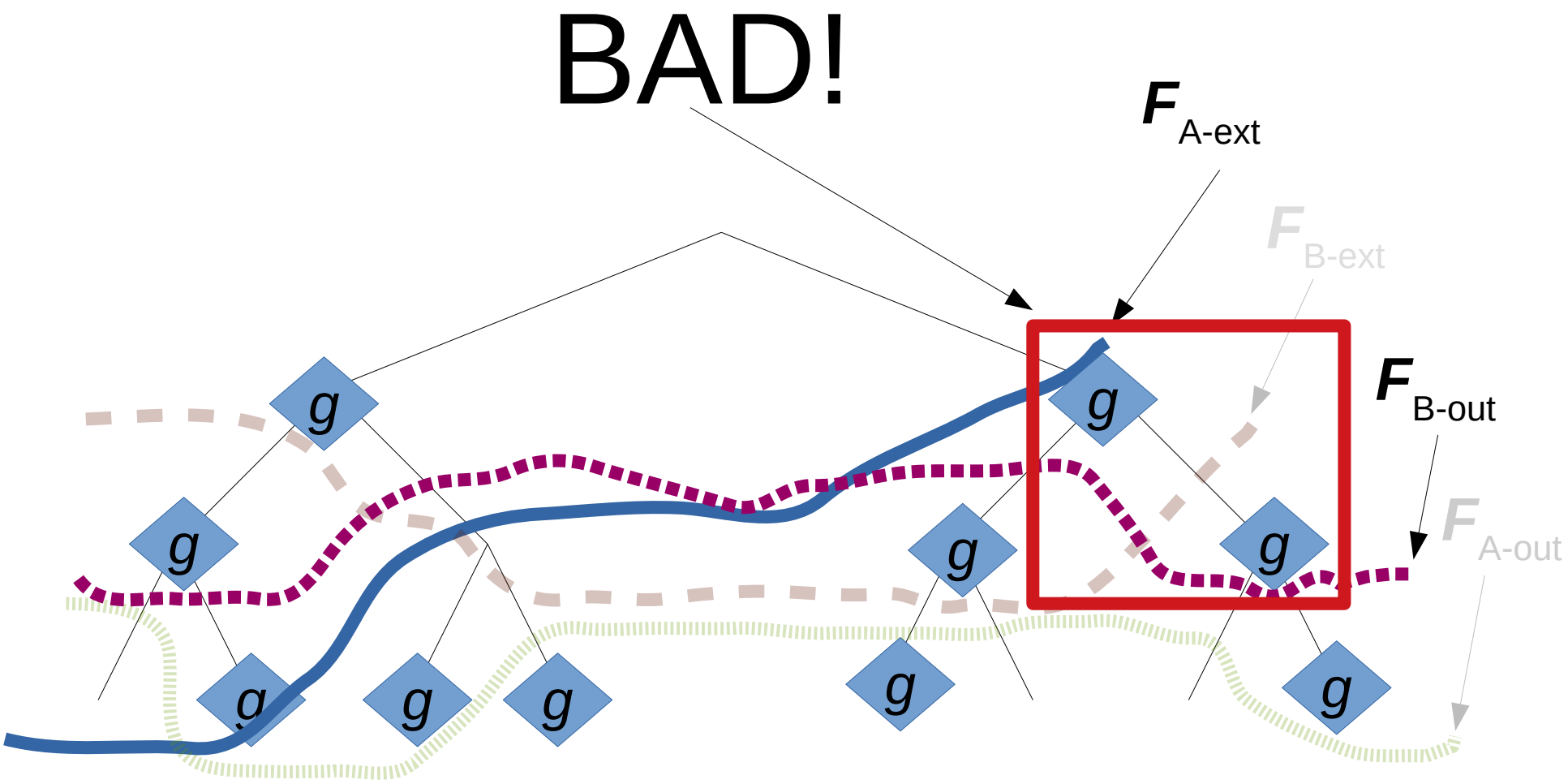
Frontiers



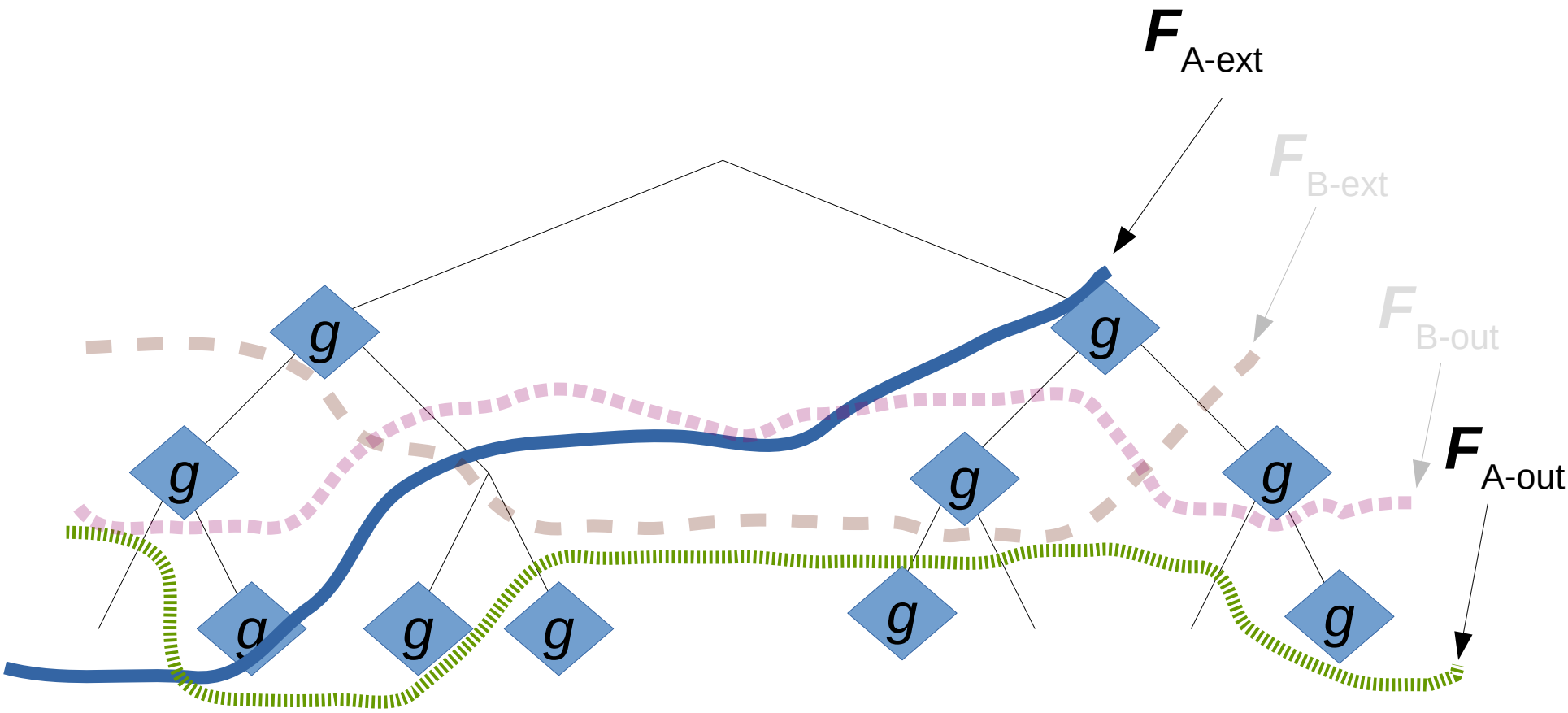
Frontiers



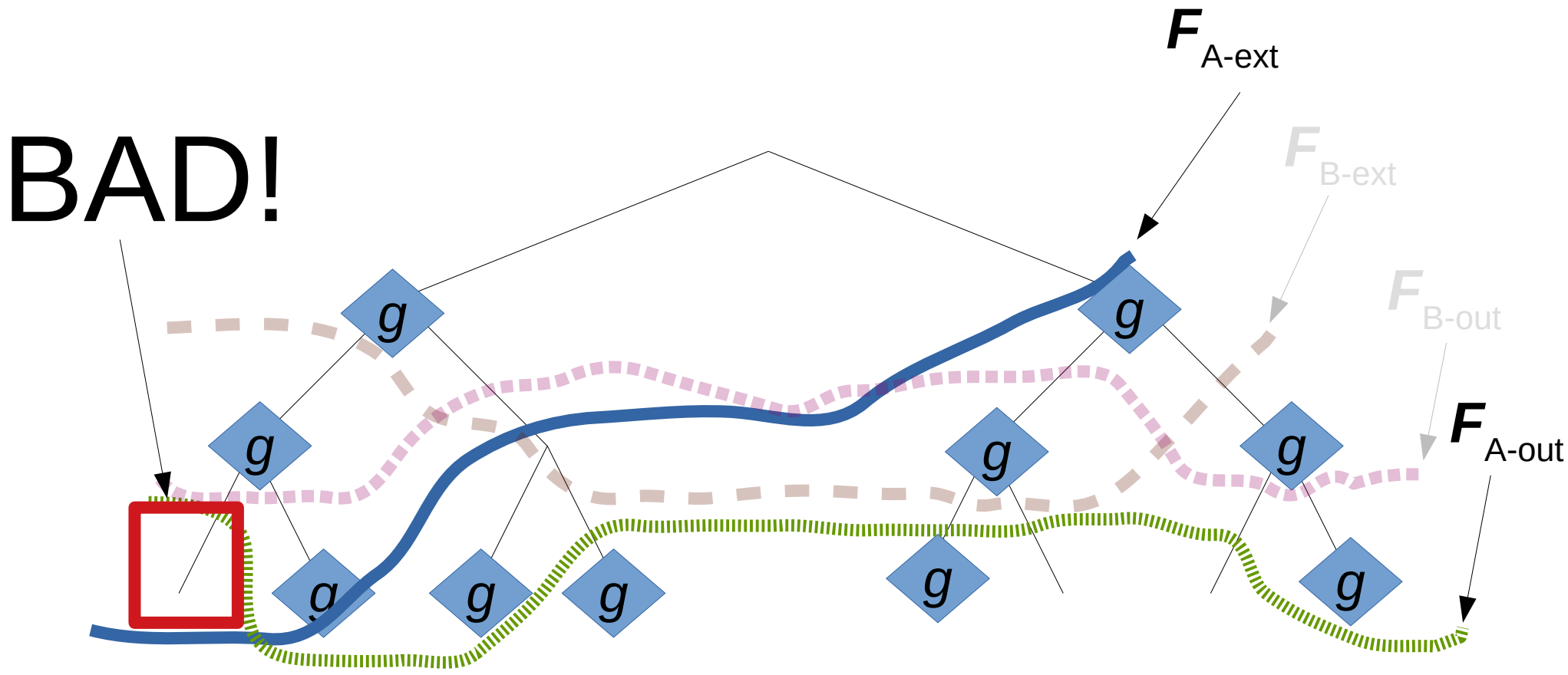
Frontiers



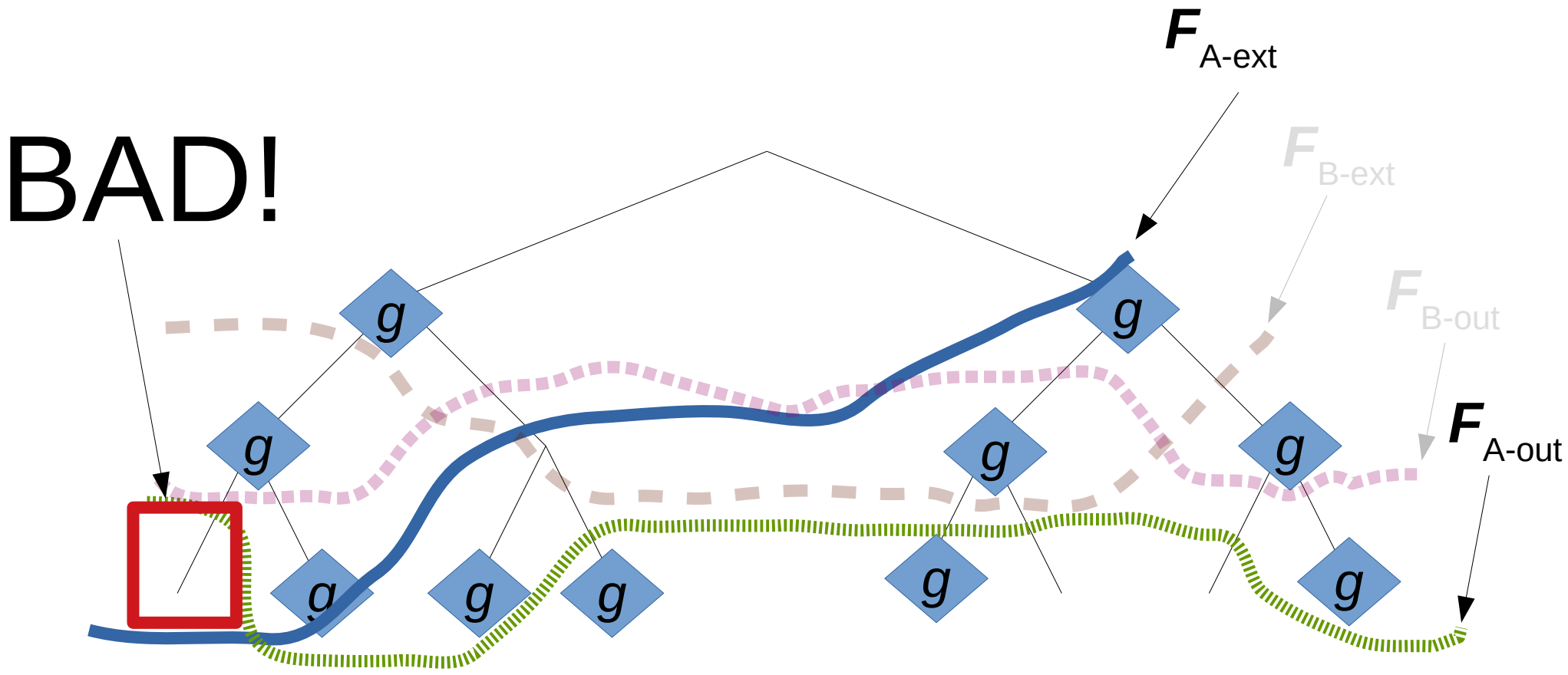
Frontiers



Frontiers



Frontiers

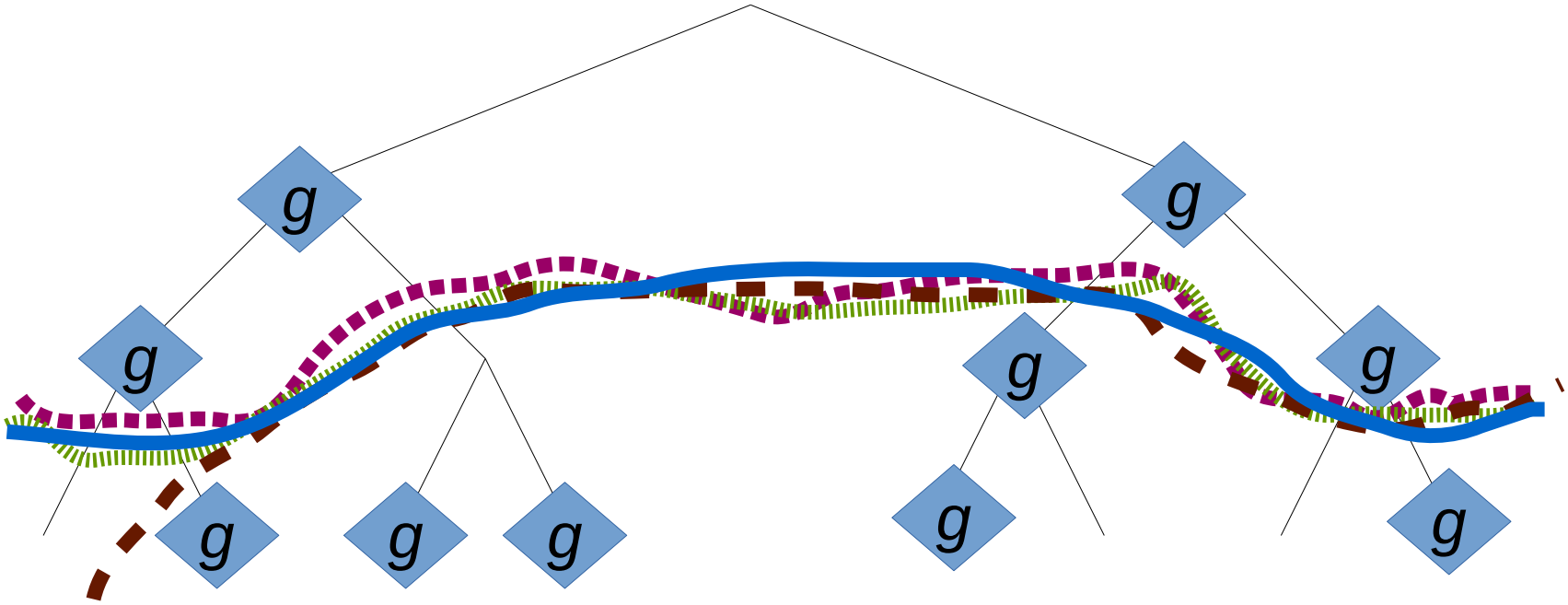


This is where we need *f* to be **non-unilateral**

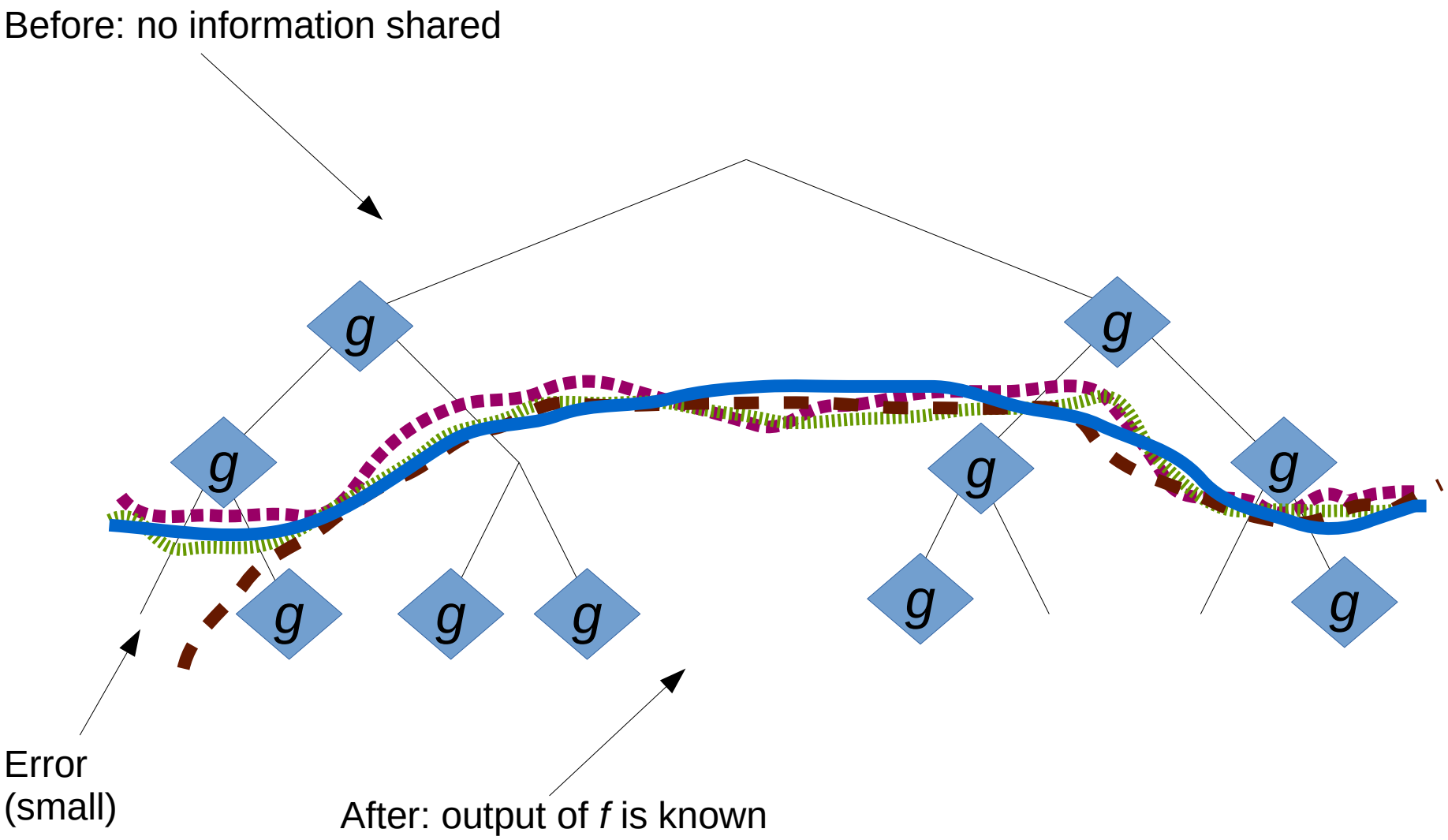
Cycle of Inequalities

- $\boldsymbol{F}_{\text{A-ext}}$ not before $\boldsymbol{F}_{\text{B-out}}$
- $\boldsymbol{F}_{\text{A-out}}$ not before $\boldsymbol{F}_{\text{A-ext}}$
- $\boldsymbol{F}_{\text{B-ext}}$ not before $\boldsymbol{F}_{\text{A-out}}$
- $\boldsymbol{F}_{\text{B-out}}$ not before $\boldsymbol{F}_{\text{B-ext}}$
- So they all happen at the same time!
- Must happen due to a call to g

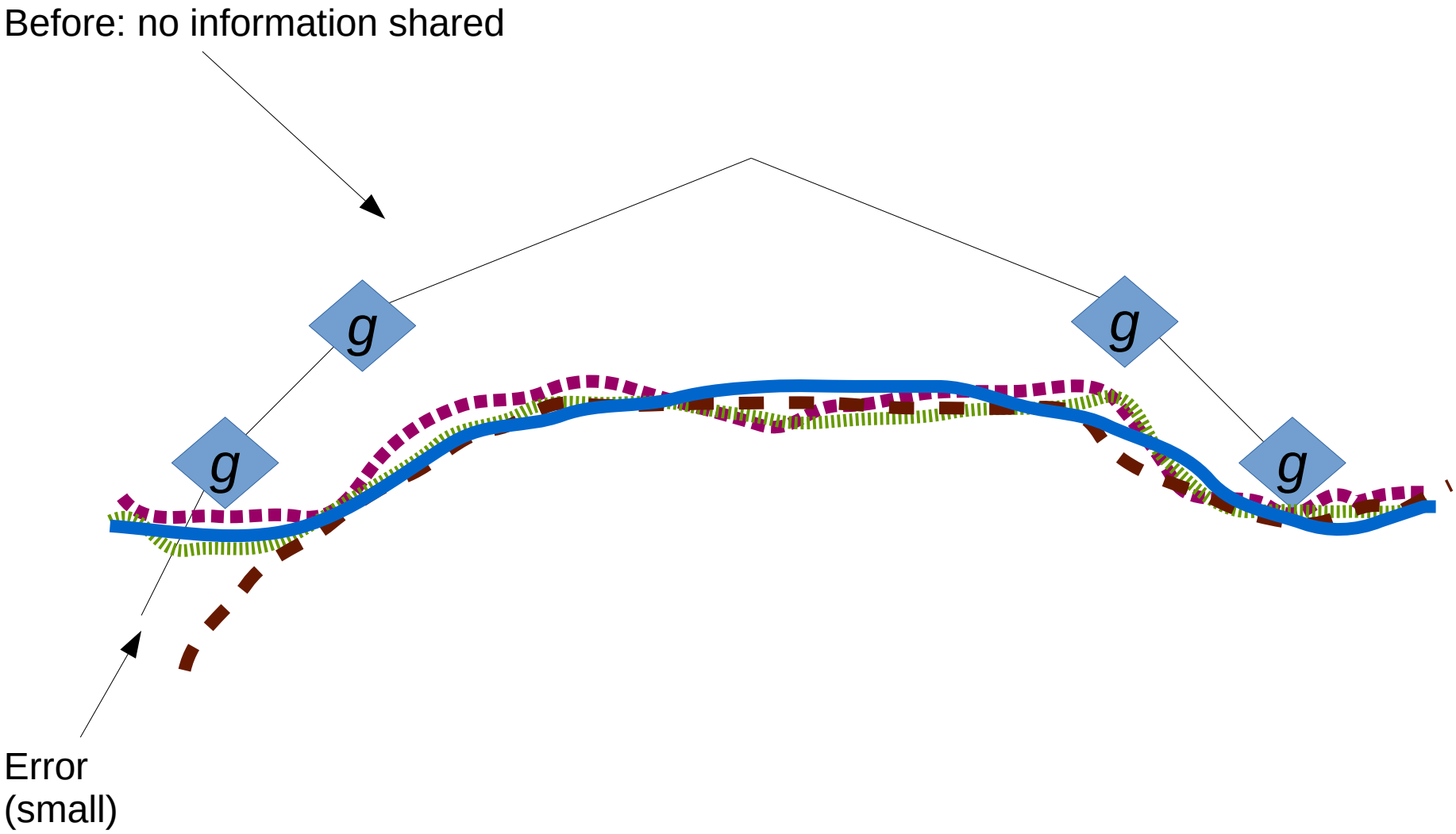
Instantaneous Property



Instantaneous Property

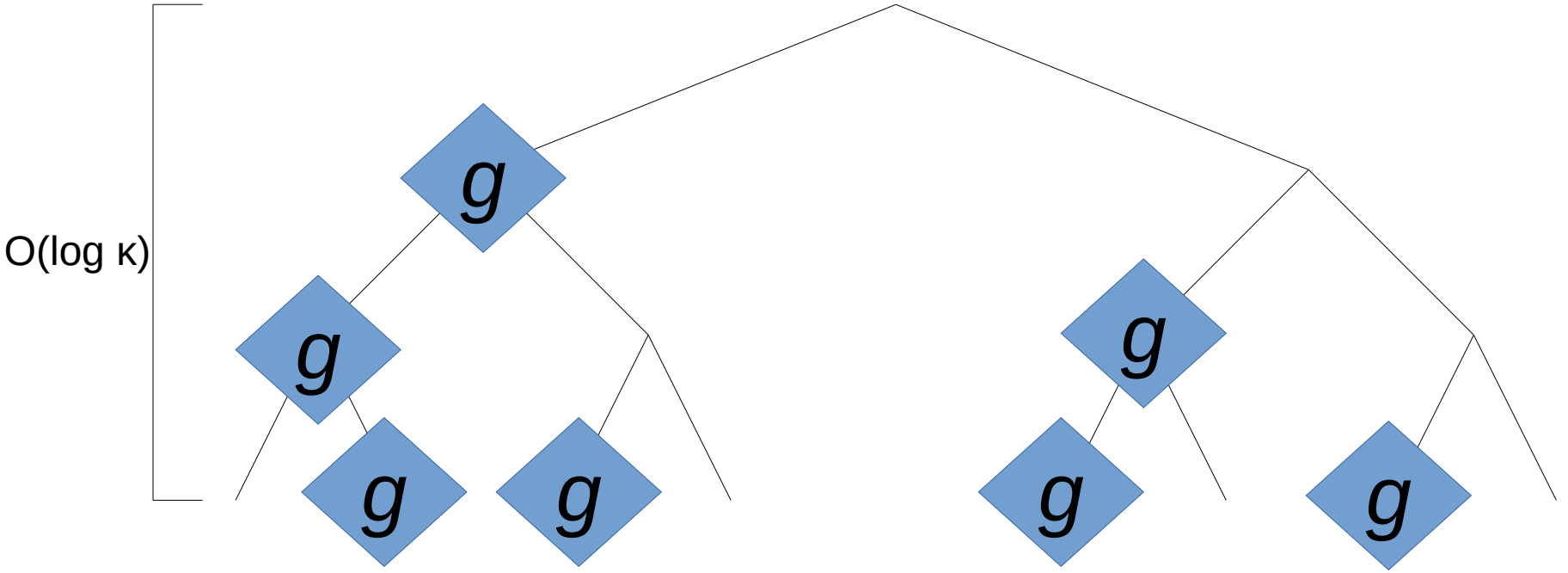


Instantaneous Property



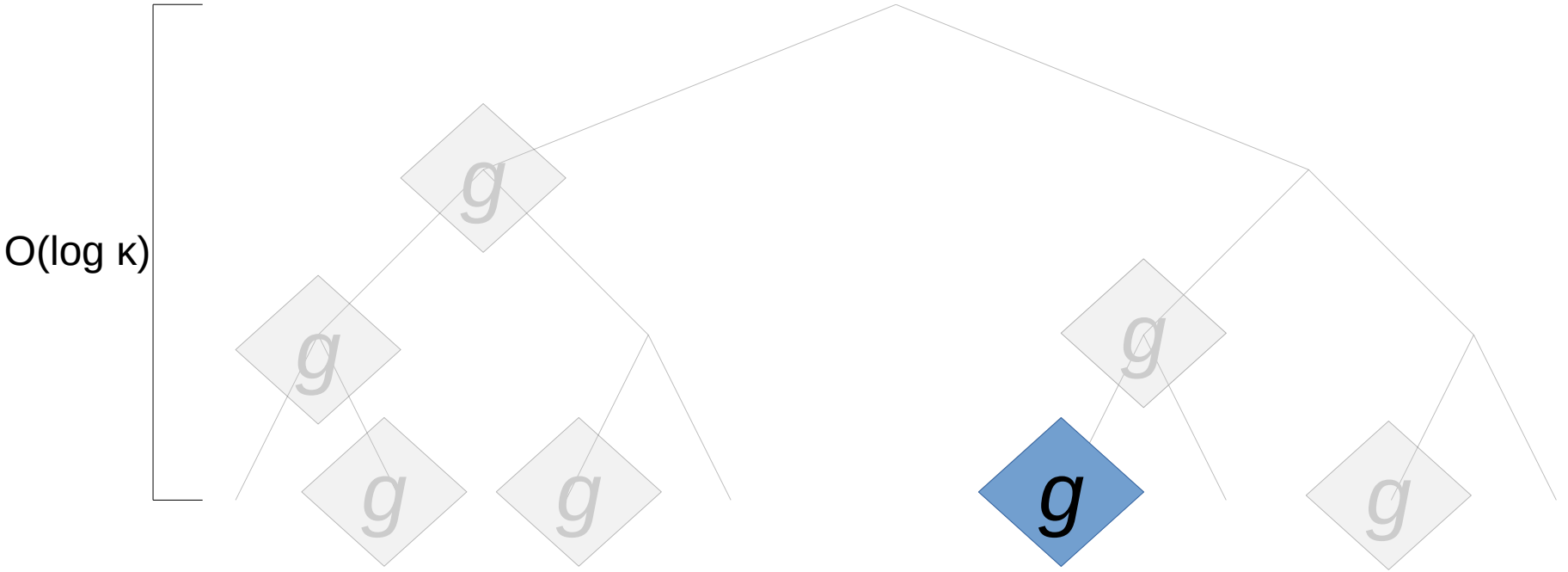
Protocol Tree

- Protocol has simulation error ε



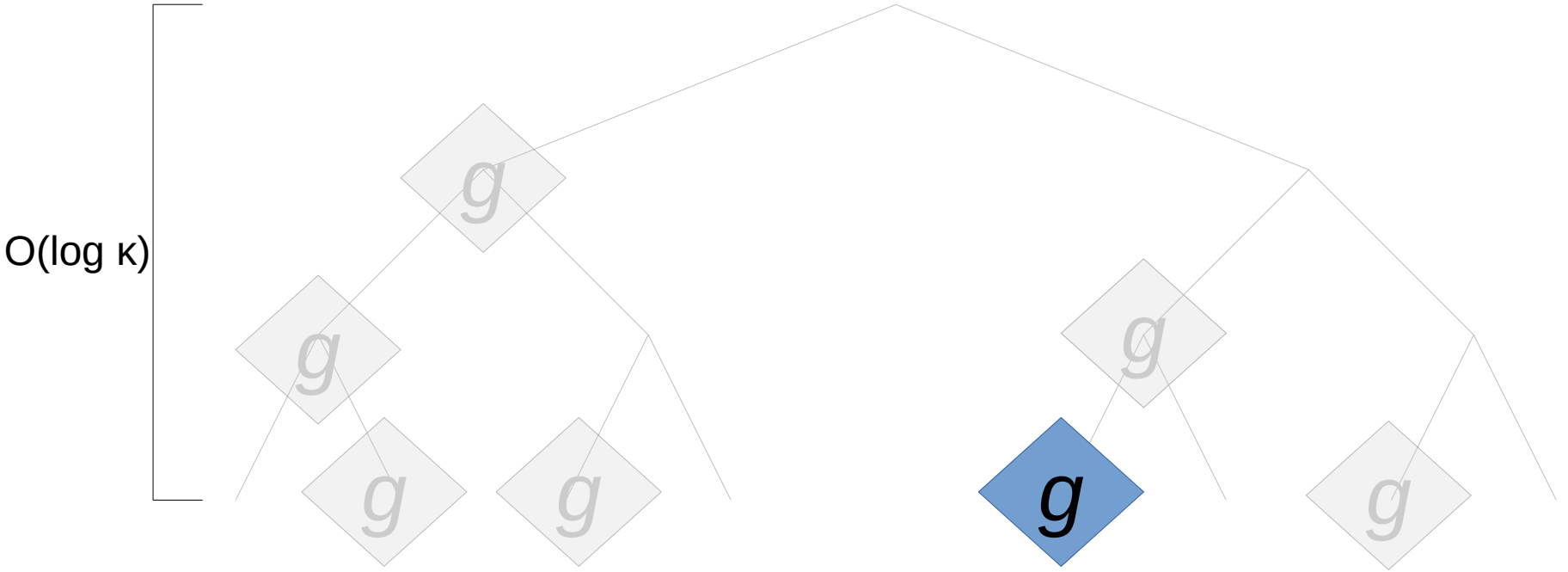
Protocol Tree

- Simulation error = ???



Protocol Tree

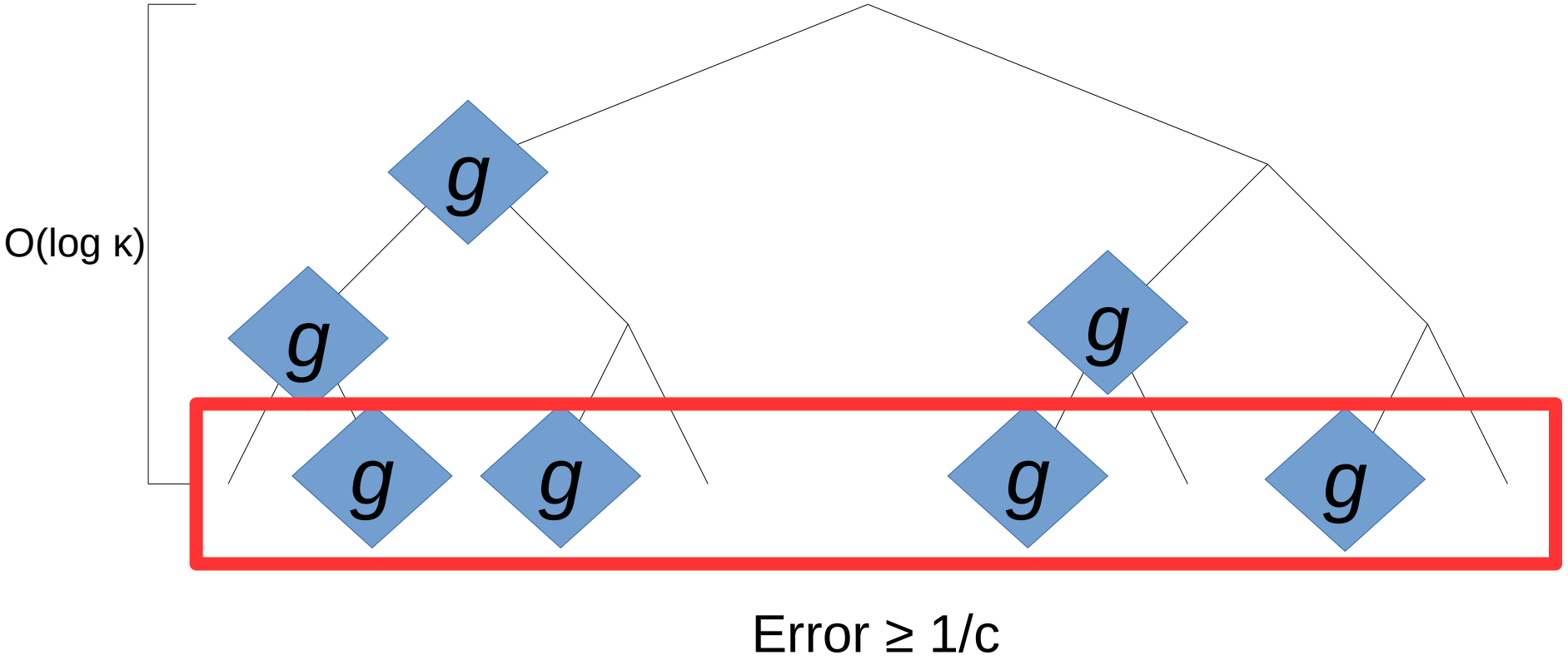
- Simulation error = ???



If the simulation error is low enough (small constant), then this is a valid protocol!

Protocol Tree

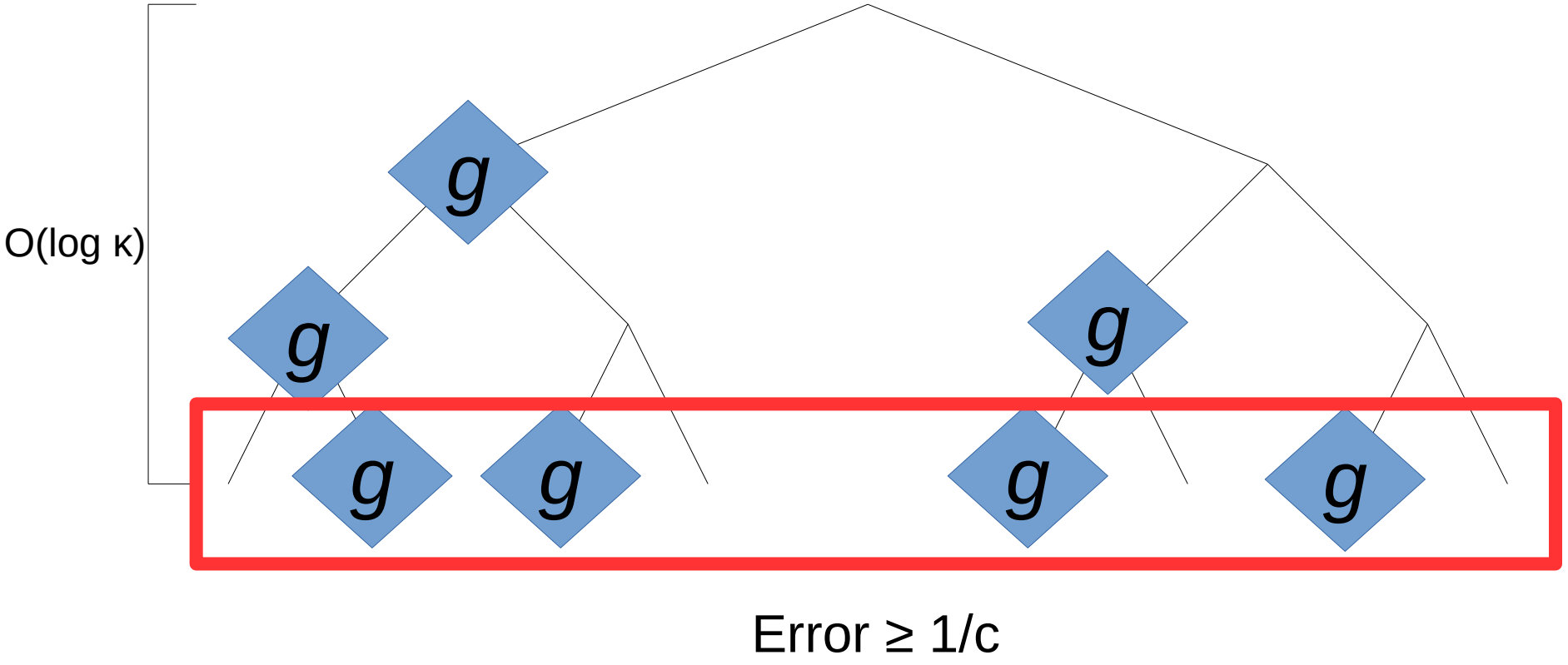
- Protocol has simulation error ε



If all of these final-round calls are not valid single-round protocols, it must be very unlikely to get to the final round!

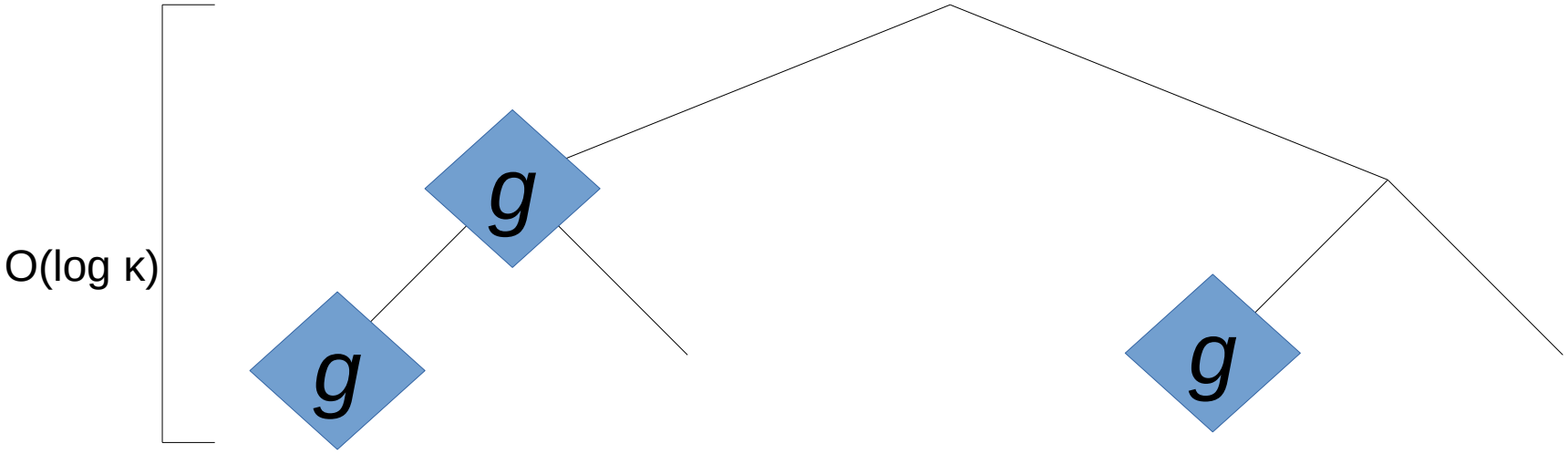
Protocol Tree

- Protocol has simulation error ε



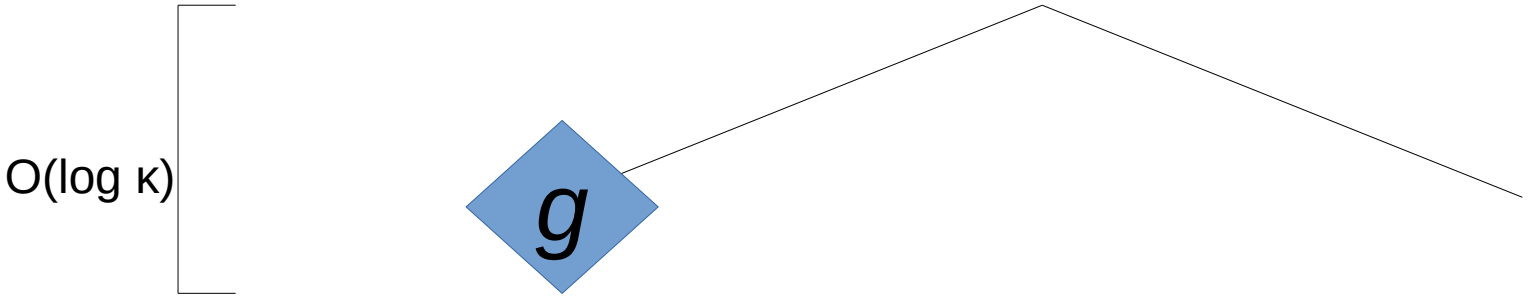
Protocol Tree

- Protocol has simulation error cε



Protocol Tree

- Protocol has simulation error $c^2\varepsilon$



Collapse to a single round

- Either we found a single-round protocol, or we repeated $O(\log \kappa)$ times
- Simulation error of the protocol truncating at the first round is $\mathbf{c^{O(\log \kappa)}\varepsilon = poly(\kappa)\varepsilon}$, which is negligible
- So this single-round protocol is a valid protocol for $f \sqsubseteq g$

Main Theorem

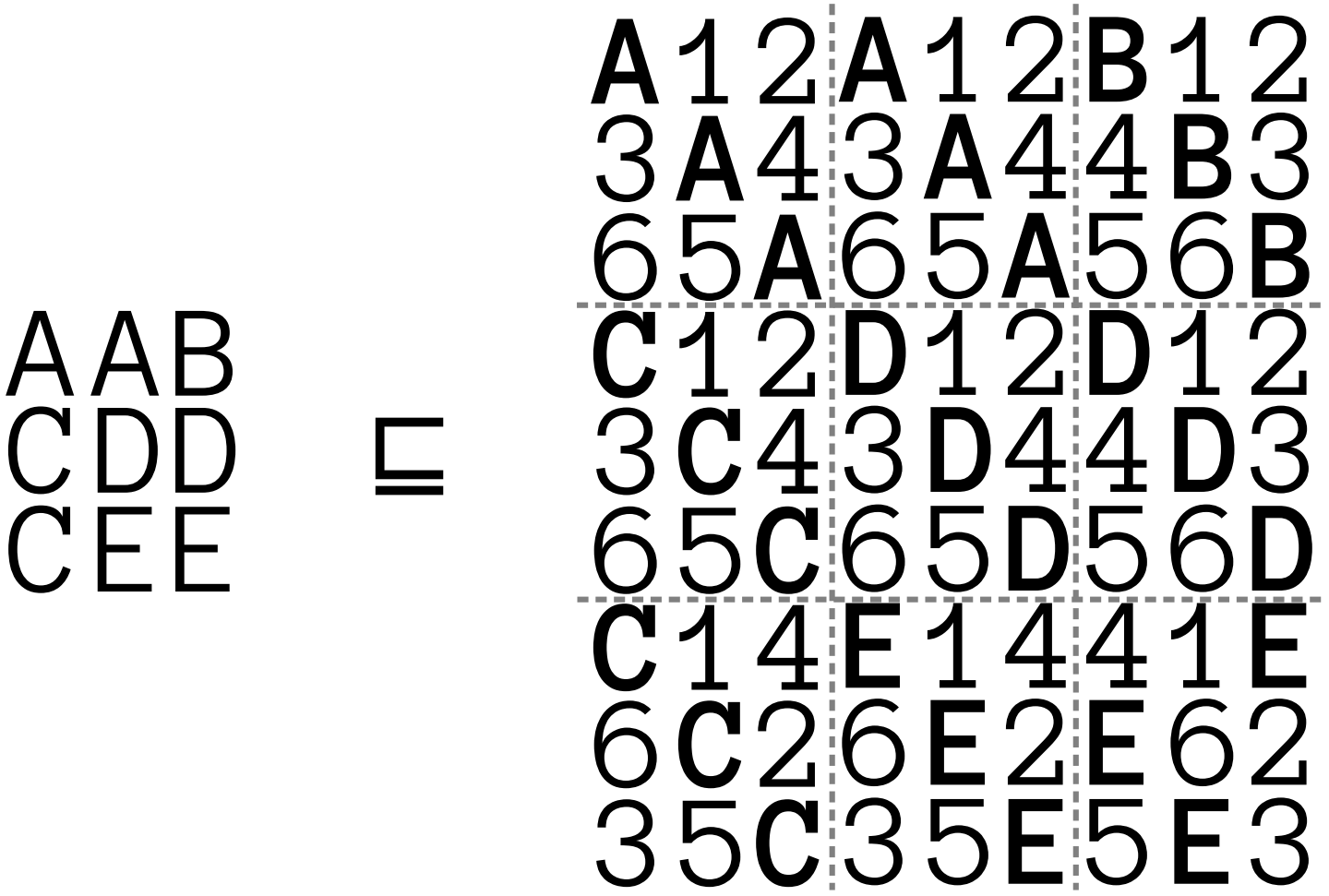
When f and g are **incomplete** and f is **non-unilateral**, the following are equivalent:

- $f \sqsubseteq g$ via a (worst-case) log-round protocol

- $f \sqsubseteq g$ via a single-round deterministic protocol

- f embeds in g

What would a protocol with $\omega(\log \kappa)$ rounds look like?



- The smallest counterexample we could find!
- No embedding (so no single round protocol)
- **Expected** number of rounds is constant (3)
- To achieve negligible error, ω(log κ) rounds are required!

Future Work: Fix the edge cases

- Unilateral functions
  - Conjecture: we only ever need to add 1 round
- Super-logarithmic protocols
  - Hard to construct examples
  - A general characterization would be interesting

Questions?